NASA CR-144990

THE USE OF GENERAL PURPOSE COMPUTER PROGRAMS TO DERIVE EQUATIONS OF MOTION FOR OPTIMAL ISOLATION STUDIES

(NASA-CR-144990) THE USE OF GENERAL PURPOSE COMPUTER PROGRAMS TO DERIVE EQUATIONS OF MOTION FOR OPTIMAL ISOLATION STUDIES Final Report (Virginia Univ.) 30 p HC $4.00 CSCL 09B G3/61
N76-23885
Unclas
40072

Final Report on

NASA Grant No. NGR 47-005-145

Supplement No. 4

Submitted by:

Walter D. Pilkey

and

Y. H. Chen

Department of Engineering Science and Systems

Research Laboratories for Engineering Sciences

University of Virginia

Charlottesville, Virginia

N.D.P.O.

Report No. ESS-4085-112-75

FOREWARD

The research described in this report (University of Virginia Project No. 4085) was performed at the University of Virginia Research Laboratories for the Engineering Sciences. W. D. Pilkey was the principal investigator. The project was performed under NASA Grant No. NGR 47-005-145 and administered at NASA Langley Research Center.

Techniques are developed which permit general purpose structural analysis computer programs to be used to generate the equations of motion necessary for limiting performance studies. The limiting performance characteristics of a system are useful in analyzing the optimal behavior of a dynamic system. In particular, the limiting performance characteristics are the essential ingredients in an efficient optimal design method for isolation systems.

INTRODUCTION

The use of conventional methods of optimal design has so far been limited to simple systems with few design parameters. This is primarily due to the overwhelming computational burdens of the methods which invariably require solving the system dynamics repeatedly. Recently a new approach that differs from the conventional methods in design methodology has been proposed and extensively explored (Reference 1, 2). The new approach, called the indirect synthesis method, selects the design parameters on the basis of a limiting performance study of the dynamic system being designed. In the process the system dynamics need be solved only once and thus the computational effort is greatly reduced. The method has been successfully applied to problems ranging from an infinite degree of freedom system with two design parameters (a beam) to a five degree of freedom system with six design parameters (an automobile model) (Reference 2). Frequently, only the limiting performance calculations are made for an isolation system as this provides useful information concerning the optimal system response. In particular, the limiting performance characteristics can be used as a yardstick for gauging the desirability of candidate system designs.

Several computer programs are now available for limiting performance studies. These are reviewed in Reference 3. Some of these programs, e.g., PERFORM (Reference 4) for transient systems and SYSLIPEC (Reference 5) for steady state systems, are virtually unlimited in terms of size and complexity of acceptable dynamic systems. PERFORM and SYSLIPEC are available through COSMIC as programs LAR-11930 and LAR-11931, respectively. The limiting performance computer programs usually require mass, stiffness, damping, and other characteristic matrices as input. The program then computes the limiting performance characteristics on the basis of this system information.

In view of the success as well as the extensive use of the finite element (FE) method in analyzing complex structures, it is important to develop the methodology necessary to couple available general purpose finite element structural programs to a limiting performance capability. This is the purpose of the work reported here. Once these general purpose programs can be used to develop input for a limiting performance computer program, then the limiting performance and indirect synthesis can be applied to any system for which the general purpose program FE is appropriate.

Approaches for coupling general purpose programs to limiting performance capabilities will be considered. Primary emphasis is given to the use of the general purpose program to develop equations of motion in a form that can be used by the limiting performance program.

BACKGROUND

Consider a linear dynamic system whose equations of motion can be written in matrix form as

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [K]\{x\} = [F]\{f\} \tag{1}$$

where

$\{x\}$ = displacement vector

$[M] = N \times N$ Mass matrix, N being the number of degrees of freedom (DOF)

$[C] = N \times N$ damping matrix

$[K] = N \times N$ stiffness matrix

$[F] = N \times L$ coefficient matrix associated with the forcing function vector $\{f\}$, L being the number of forcing functions.

The optimal design problem is to choose portions of the system so that some index of performance is minimzed (or maximized) and certain constraints

are satisfied. It is assumed that both the performance index (ψ) and the constraint functions $C_k(x,t)$ on the response variables involve peak response variables, e.g., maximum stresses, displacements, accelerations. The performance index and the constraint functions can be written in the forms

$$\psi = \max_r \max_t |h_r|, \quad r = 1,2, \ldots R$$
$$C_k^L \le C_k \le C_k^U, \quad k = 1,2, \ldots K \tag{2}$$

where h_r's are the R response functions, and C_k^U, C_k^L are the prescribed upper and lower bounds of the kth constraint. We will restrict our consideration to cases wherein h_r and C_k are linear functions of the response (or state) variables.

The first step in using the indirect synthesis method is to calculate the limiting performance. Those portions of the system to be designed are replaced by generic (or control) forces $\{u(t)\}$, so that the equations of motion now become

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [\bar{K}]\{x\} + [V]\{u\} = [F]\{f\} \tag{3}$$

where $[\bar{K}]$ is the new stiffness matrix, which omits any stiffness contributions from the design elements, and $[V]$ is the N x J coefficient matrix associated with the generic forces $\{u\}$, J being the number of controllers. Such a general dynamical system is shown schematically in Fig. 1. Two types of elements are considered; structural elements and isolator elements. There may be any number of each; in particular M structural elements and J isolator elements could be interconnected in an arbitrary fashion. A structural element may represent a discrete mass point, a rigid body of distributed mass, or a flexible structure such as a framework or a shell. The isolator elements, similarly, can represent either simple mechanisms without mass or models of more complicated

devices. In general, the structural elements constitute the prescribed portions of the system (i.e., the base structure and the elements to be isolated) and the isolator elements are to be chosen in accordance with the design objectives. Note also that the control forces $\{u(t)\}$, being considered as explicit functions of time, do not contribute to the nonlinearity of the equations of motion even though they may have replaced nonlinear portions of the system, e.g. a nonlinear spring connection.

Once Eqs. (3) are established, and the performance index ψ and the constraints are placed in the form of Eqs. (2), then existing limiting performance capabilities can be used to obtain the limiting performance characteristics which are subsequently employed in selecting the design parameters.

As mentioned in Reference 1, the problem of establishing the coefficient matrices $[M]$, $[C]$, $[\bar{K}]$, $[V]$, and $[F]$ in Eq. (3) for a particular dynamic system can be avoided by employing a general purpose dynamic program to generate impulse responses at isolator attachment points and then constructing h_r, C_k with Duhamel (convolution) integrals. That is, since the system under consideration is linear, the response of the system excited by arbitrary forces u_j can be obtained by superimposing responses to unit impulses placed sequentially at each of the isolator attachment points, i.e.

$$
\begin{aligned}
h_r(t) &= h_{r0}(t) + \sum_{j=1}^{J} \int_0^t g_{rj}(t - \tau)u_j(\tau)d\tau \\
C_k(t) &= C_{k0}(t) + \sum_{j=1}^{J} \int_0^t g_{kj}(t - \tau)u_j(\tau)d\tau
\end{aligned}
\tag{4}
$$

where g_{rj} and g_{kj} are the appropriate system responses to a unit impulse

at the attachment point of the jth isolator, and $h_{ro}(t)$, $C_{k0}(t)$ are the responses to the L inputs $f_{\ell}(t)$. The g_{rj} and g_{kj} must be generated by the structural dynamics program. The advantage of this approach lies in the fact that an existing general-purpose structural dynamics code will be used to perform all of the system dynamics required for the limiting performance problem. However, in this work we are concerned with the situation whereby the system dynamics are to be solved by a limiting performance capability and not by the general purpose program. The general purpose program will be used to generate the coefficient matrices of Eqs. (3).

To illustrate the problems involved in preparing the equations for a limiting performance solution, consider the simple example of a three degrees of freedom spring-mass system as shown in Fig. 2a. Let it be desired to select the spring constant k_2 such that the maximum acceleration transmitted to any mass is minimzed while the three rattlespaces satisfy prescribed constraints. The k_1, k_2 spring rates remain at their prescribed values. For this example, the response functions that make up the performance index are

$$h_1 = \ddot{z}_1,\ h_2 = \ddot{z}_2,\ h_3 = \ddot{z}_3$$

and the constraint functions are

$$C_1 = z_1 - z_2,\ C_2 = z_2 - z_3,\ C_3 = z_3 - f$$

Therefore, Eqs. (2) become

$$\psi = \max_r \max_t |\ddot{z}_r|,\quad r = 1,2,3$$

or

$$\psi = \max[\max_t |\ddot{z}_1|,\ \max_t |\ddot{z}_2|\ ,\ \max_t |\ddot{z}_3|]$$

$$C_1^L \leq |z_1 - z_2| \leq C_1^U$$

$$C_2^L \leq |z_2 - z_3| \leq C_2^U$$

$$C_3^L \leq |z_3 - f| \leq C_3^U$$

where C_1^L, C_1^U, $\ldots C_3^U$ are prescribed constants, i.e. prescribed constraint bounds. The equations of motion for this system are easily found to be

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{Bmatrix} + \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1+k_2 & -k_2 \\ 0 & -k_2 & k_2+k_3 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = - \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \ddot{f} \quad (5)$$

where $x_i = z_i - f$, $i = 1,2,3$. Now replace the middle spring by a generic force u_2 (Fig. 2b). This results in a new set of equations

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{Bmatrix} + \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 & 0 \\ 0 & 0 & k_3 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} u_2 = - \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \ddot{f} \quad (6)$$

Equations (6) can be obtained either by applying the conditions of equilibrium to a configuration taken from Fig. 2b or by directly making the substitution $u_2 = k_2(z_2 - z_3) = k_2(x_2 - x_3)$ in each row of Eq. (5), which amounts to setting $k_2 = 0$ and taking the coefficients of k_2 in row 2 of the [K] matrix as the column of the [V] matrix. Obviously, the first approach becomes impractical when the system is complicated or when the system is being analyzed by FE Codes since it contains unknown elements. In the following section, the second approach will be generalized and discussed further.

DEVELOPMENT OF THE EQUATIONS OF MOTION

As mentioned, in order to use most existing limiting performance computer capabilities, we must express the FE system equations of motion

in the form of Eq. (3) rather than in the usual FE form of Eq. (1). Standard FE codes will always generate the mass matrix [M], the damping matrix [C], and the coefficient matrix [F] which can be fed into the limiting performance codes with little or no modification. Thus it remains to create the new stiffness matrix $[\bar{K}]$ and the controller matrix [V]. Here we describe two techniques.

Technique 1

The replacement of an isolator to element k_i by a control force u_i amounts to substituting in the equations of motion the relations

$$u_i = k_i \Sigma a_j x_j \tag{7}$$

where the summation is over the degrees of freedom that are connected with isolator k_i and the a_j's are the kinematical factors that are associated with each degree of freedom. This implies that isolator k_i no longer contributes to the assembly stiffness matrix and instead is replaced by an additional matrix [V], whose individual columns consist of the coefficients of the term $k_i \Sigma a_j x_j$ in the equations of motion. With the aid of the element stiffness matrix written in assembly (global) coordinates, we can easily obtain the new stiffness matrix $[\bar{K}]$ and the controller matrix [V] from [K]. From the node numbering scheme of a FE code, we know which elements of [K] are contributed by isolator k_i so that the new stiffness matrix $[\bar{K}]$ is obtained by simply removing those contributions. In the FE program, set the spring elements to be optimized temporarily equal to zero and the desired $[\bar{K}]$ is produced by printing the stiffness matrix. The controller matrix [V] is formed by taking as its column the coefficients of k_i in a row (or column) of [K], the row (or column) number being that of a degree of freedom which is enacted by the isolator.

Technique 2

Rewrite Eq. (3) as

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [\bar{K}]\{x\} = [F]\{\ddot{f}\} - [V]\{u\}$$

$$= [F]\{\ddot{f}\} - \begin{bmatrix} V_{11} & V_{12} & \cdots & V_{1J} \\ V_{21} & V_{22} & \cdots & V_{2J} \\ \vdots & & & \vdots \\ V_{N1} & V_{N2} & \cdots & V_{NJ} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ \vdots \\ u_J \end{Bmatrix}$$

$$= [F]\{\ddot{f}\} - \begin{Bmatrix} V_{11} \\ V_{21} \\ \vdots \\ V_{N1} \end{Bmatrix} u_1 - \begin{Bmatrix} V_{12} \\ V_{22} \\ \vdots \\ V_{N2} \end{Bmatrix} u_2 - \cdots \begin{Bmatrix} V_{1J} \\ V_{2J} \\ \vdots \\ V_{NJ} \end{Bmatrix} u_J \tag{8}$$

The $[\bar{K}]$ is found as in technique 1. The columns of [V] are obtained by noting that the elements of [V] are simply the influence coefficients of the "loads" {u}. Thus to obtain the influence coefficients of a control force u_i (i.e. to obtain the ith column of [V]), set all u's equal to zero, save u_i which is set equal to 1 (Fig. 3). The resulting load vector generated by FE program is the desired column. In practice, one must be careful to properly account for the constraints when generating the $[\bar{K}]$, [V] matrices from a FE code, i.e. it is necessary to ascertain whether the matrices are printed before or after the displacement constraints have been applied to the problem.

Three Degree of Freedom System Example

Consider the problem of Fig. 2 which was studied above. To treat a particular case, let $m_1 = 0.5$, $m_2 = 1.5$, $m_3 = 2.5$, $k_1 = 10$, $k_2 = 20$, $k_3 = 30$. Substitution of these values in Eq. (5) gives

$$[M] = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 2.5 \end{bmatrix}, \quad [K] = \begin{bmatrix} 10 & -10 & 0 \\ -10 & 30 & -20 \\ 0 & -20 & 50 \end{bmatrix} \tag{9}$$

In terms of global coordinates the stiffness $[K_2]$ of the middle spring can be written as

$$[K_2] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 20 & -20 \\ 0 & -20 & 20 \end{bmatrix} \tag{10}$$

It follows from Eq. (10) that the stiffness of the middle spring contributes to K_{22}, K_{23}, K_{32}, and K_{33}. Removing these contributions from [K] of Eq. (9) results in

$$[\bar{K}] = \begin{bmatrix} 10 & -10 & 0 \\ -10 & 10 & 0 \\ 0 & 0 & 30 \end{bmatrix} \tag{11}$$

and from the coefficients of elements in row 2 of $[K_2]$ we find

$$[V] = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$$

which implies that $u_2 = k_2(z_2 - z_3) = k_2(x_2 - x_3)$.

To generate the desired matrices [M], $[\bar{K}]$, and [V] with a general purpose program using the second technique, first compute [M] as though a dynamics problem were to be run. Then run a static case to compute $[\bar{K}]$ and [V] applying unit loads as shown in Fig. 4.

PERFORMANCE INDEX AND CONSTRAINTS

Now that the equations of motion are in the proper form for a limiting performance study, we proceed to establish the performance index and constraints.

Any linear combination of accelerations, velocities, displacements, or control forces can be used as a performance index or constraint function, i.e.

$$\{h\} \text{ or } \{C\} = [P]\{\ddot{x}\} + [Q]\{\dot{x}\} + [R]\{x\} + [S]\{u\} + [T]\{f\} \quad (12)$$

where [P], [Q], [R], [S] and [T] are prescribed coefficient matrices.

The coefficient matrices in Eq. (12) can be easily formed when using a FE code. For example, consider the discretized system shown in Fig. 5. At each node, six degrees of freedom (DOF)(3 translational and 3 rotational) are specified. Suppose in Fig. 5 that the accelerations along the y coordinate at node i and node j are to be constrained such that

$$|a\ddot{y}_i + b\ddot{y}_j| \leq d \quad (13)$$

where a, b, d are prescribed constants. To place the constraint function $a\ddot{y}_i + b\ddot{y}_j$ in the form of Eq. (12), we need only input the node numbers, the nodal DOF, and the coefficients a, b, e.g. (i,2,a) and (j,2,b) where the 2 stands for the y degrees of freedom. In most general purpose FE codes, there is a printable connectivity array that identifies the nodal DOF to be constrained in terms of the independent assembly DOF. Suppose that (i,2) and (j,2) correspond to Ith and Jth assembly DOF. Then an appropriate kth row of the [P] matrix in Eq. (12) will consist of the elements

$$P_{kI} = a,\ P_{kJ} = b,\ k \text{ being the row number}$$
$$P_{k\ell} = 0 \text{ for all } \ell \neq I,J. \quad (14)$$

For this example the elements of the kth rows of the matrices [Q], [R], [S], [T] are all zero since the constraint function of Eq. (13) contains only accelerations. Similar reasoning permits Eq. (12) to be formed for

any acceptable type of performance index and constraint function.

Although we have chosen to describe a dynamic system by second order differential equations, Eqs. (3), some limiting performance programs work with first order differential equations. The second order equations are converted into a set of 2N first order linear differential equations

$$\{\dot{s}\} = [A]\{s\} + [B]\{\bar{u}\} + [D]\{\bar{f}\} \tag{15}$$

where

$$\{s\} = \begin{Bmatrix} \{\dot{x}\} \\ --- \\ \{x\} \end{Bmatrix}$$

is a state vector with 2N components, and the coefficient matrices [A], [B], [D] are determined as functions of the coefficient matrices of Eq. (3). Also, instead of Eq. (12), the performance index and constraints are input to the limiting performance programs in the form

$$\{h\} \text{ or } \{C\} = [E]\{s\} + [G]\{u\} + [H]\{f\} \tag{16}$$

where [E], [G], and [H] are prescribed coefficient matrices. Although the formation of Eq. (15) using Eq. (3) is a familiar manipulation, it is useful to consider in detail how Eq. (16) is obtained from Eq. (12). This is accomplished by solving Eq. (3) for $\{\ddot{x}\}$ and placing the result in Eq. (12) to give

$$\{h\} \text{ or } \{C\} = ([Q] - [P]\left[\frac{C}{M}\right])\{\dot{x}\} + ([R] - [P]\left[\frac{\bar{K}}{M}\right])\{x\}$$

$$+ ([S] - [P]\left[\frac{V}{M}\right])\{u\} + ([T] + [P]\left[\frac{F}{M}\right])\{f\} \tag{17}$$

A comparison of Eq. (15) with Eq. (17) shows that

$$[E] = \left[[Q] - [P]\left[-\frac{C}{M}-\right] \;\vdots\; [R] - [P]\left[-\frac{\bar{K}}{M}-\right] \right]$$

$$[G] = [S] - [P]\left[-\frac{V}{M}-\right] \tag{18}$$

and

$$[H] = [T] + [P]\left[-\frac{F}{M}-\right]$$

USE OF SAP IV

To implement the techniques described here the SAP IV computer program was selected. SAP IV (Reference 6) is a popular general purpose program that can be altered locally with relative ease. The program required only slight modification in order to print out the desired matrices for Eqs. (3). In the following examples, all springs were input to SAP IV as elastic beam elements with appropriately chosen moduli and dimensions to secure the correct spring constants. The program allows for slave degrees of freedom to treat the rigid beams. It, however, ignores any loads applied at the slave nodes and thus these loads must be replaced by their static equivalents at the master node.

Three Degree of Freedom System Example

SAP IV was used to form the [M], [$\bar{K}$], and [V] matrices for the three degree of freedom example of Fig. 2, with mass and spring values of the previous example. The results are given in Fig. 6. In its dynamic mode of operation, SAP IV computed [M]. Then, in a static mode [$\bar{K}$] and [V] were found by applying the unit loads of Fig. 4.

Five Degree of Freedom System Example

As a more complicated example consider the automobile model of Fig. 7, which is considered in Reference 5. The equations of motion for the configuration shown are

$$m_1\ddot{z}_1 + k_1(z_1 - f) - k_2(z_2 - z_1 - \frac{\ell}{3}\theta) = 0$$

$$m_2\ddot{z}_2 + k_2(z_2 - z_1 - \frac{\ell}{3}\theta) - k_3(z_3 - z_2) + k_4(z_2 - z_4 + \frac{2\ell}{3}\theta) = 0$$

$$I\ddot{\theta} - k_2(z_2 - z_1 - \frac{\ell}{3}\theta)\frac{\ell}{3} + k_4(z_2 - z_4 + \frac{2\ell}{3}\theta)\frac{2\ell}{3} = 0$$

$$m_3\ddot{z}_3 + k_3(z_3 - z_2) = 0$$

$$m_4\ddot{z}_4 - k_4(z_2 - z_4 + \frac{2\ell}{3}\theta) + k_5(z_4 - f) = 0$$

or, in matrix form

$$\begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & m_3 & 0 \\ 0 & 0 & 0 & 0 & m_4 \end{bmatrix} \begin{Bmatrix} \ddot{z}_1 \\ \ddot{z}_2 \\ \ddot{\theta} \\ \ddot{z}_3 \\ \ddot{z}_4 \end{Bmatrix}$$

$$+ \begin{bmatrix} k_1+k_2 & -k_2 & \frac{\ell}{3}k_2 & 0 & 0 \\ -k_2 & k_2+k_3+k_4 & -\frac{\ell}{3}k_2 + \frac{2\ell}{3}k_4 & -k_3 & -k_4 \\ \frac{\ell}{3}k_2 & -\frac{\ell}{3}k_2 + \frac{2\ell}{3}k_4 & \frac{\ell^2}{9}k_2 + \frac{4\ell^2}{9}k_4 & 0 & -\frac{2\ell}{3}k_4 \\ 0 & -k_3 & 0 & k_3 & 0 \\ 0 & -k_4 & -\frac{2\ell}{3}k_4 & 0 & k_4+k_5 \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \\ \theta \\ z_3 \\ z_4 \end{Bmatrix}$$

$$= \begin{bmatrix} k_1 \\ 0 \\ 0 \\ 0 \\ k_5 \end{bmatrix} f \qquad (21)$$

If k_2 is replaced by u_2 so that

$$u_2 = k_2(z_2 - z_1 - \frac{\ell}{3}\theta)$$

then we have

$$[\bar{K}] = \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 \\ 0 & k_3+k_4 & \frac{2\ell}{3}k_4 & -k_3 & -k_4 \\ 0 & \frac{2\ell}{3}k_4 & \frac{4\ell^2}{9}k_4 & 0 & -\frac{2\ell}{3}k_4 \\ 0 & -k_3 & 0 & k_3 & 0 \\ 0 & -k_4 & -\frac{2\ell}{3}k_4 & 0 & k_4+k_5 \end{bmatrix}, \quad [V] = \begin{bmatrix} -1 \\ +1 \\ \frac{-\ell}{3} \\ 0 \\ 0 \end{bmatrix}, \quad \{u\} = u_2 \quad (22)$$

If in addition, k_4 is replaced by a control force u_4 so that

$$u_4 = k_4(z_2 - z_4 + \frac{2\ell}{3}\theta)$$

then

$$[\bar{K}] = \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 \\ 0 & k_3 & 0 & -k_3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -k_3 & 0 & k_3 & 0 \\ 0 & 0 & 0 & 0 & k_5 \end{bmatrix}, \quad [V] = \begin{bmatrix} -1 & 0 \\ 1 & 1 \\ \frac{-\ell}{3} & \frac{2\ell}{3} \\ 0 & 0 \\ 0 & -1 \end{bmatrix}, \quad \{u\} = \begin{Bmatrix} u_2 \\ u_4 \end{Bmatrix} \quad (23)$$

As a numerical example take $m_1 = m_4 = 2$, $m_2 = 8$, $I = 0.5$, $m_3 = 1$, $k_1 = k_5 = 1$, $k_2 = k_4 = 2$, $k_3 = 4$, $\ell = 12$, so that

$$[M] = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 0 \\ 0 & 0 & .5 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} , \quad [K] = \begin{bmatrix} 3 & -2 & 8 & 0 & 0 \\ -2 & 8 & 8 & -4 & -2 \\ 8 & 8 & 160 & 0 & -16 \\ 0 & -4 & 0 & 4 & 0 \\ 0 & -2 & -16 & 0 & 3 \end{bmatrix} \tag{24}$$

If both k_2 and k_4 are replaced, then from Eq. (23) we have

$$[\bar{K}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & -4 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} , \quad [V] = \begin{bmatrix} -1 & 0 \\ 1 & 1 \\ -4 & 8 \\ 0 & 0 \\ 0 & -1 \end{bmatrix}$$

In using SAP IV to generate these matrices, we first ran it in a dynamic mode to obtain [M] and [K] (displayed in Fig. 8) and then ran it as a static case by applying unit loads as shown in Fig. 9. This latter run yields the desired $[\bar{K}]$ and [V] matrices (Fig. 10).

SUMMARY

The techniques have been established whereby a general purpose structural analysis program can be used to form the equations required for available limiting performance capabilities (references 3,4,5). These techniques would permit a limiting performance study to be conducted on an arbitrary structural or mechanical system, which normally would involve finite element modeling for an analysis. The SAP IV general purpose computer program was employed to carry out a number of numerical examples to verify the formulation presented here.

REFERENCES

1. E. Sevin, W. D. Pilkey, Optimum Shock and Vibration Isolation, Shock and Vibration Information Center, Wash. D. C., 1971.

2. W. D. Pilkey, Y. H. Chen, A. J. Kalinowski, "Indirect Synthesis of Multidegree of Freedom Transient Systems." J. of Optimization Theory and Applications, to appear.

3. W. D. Pilkey, P. Smith, "Limiting Performance" in Shock and Vibration Computer Programs, Shock and Vibration Information Center, Wash. D. C., 1975.

4. W. D. Pilkey, B. P. Wang, Y. Yoo, and B. Clark, "PERFORM - A Performance Optimizing Computer Program for Dynamic Systems Subject to Transient Loadings," NASA CR-2268, 1973.

5. W. D. Pilkey, B. P. Wang, "SYSLIPEC - A Limiting Performance Computer Program for Linear Dynamic Systems Subject to Steady-State Sinusoidal Loading," Research Laboratories for the Engineering Sciences Report No. ESS-4085-109-73, University of Virginia, NASA Grant No. NGR 47-005-145, 1973.

6. K. J. Bathe, E. L. Wilson, and F. E. Peterson, "SAP IV, A Structural Analysis Program for Static and Dynamic Response of Linear Systems," Report No. EERC 73-11, University of California, Berkeley, June 1973, Revised April 1974.

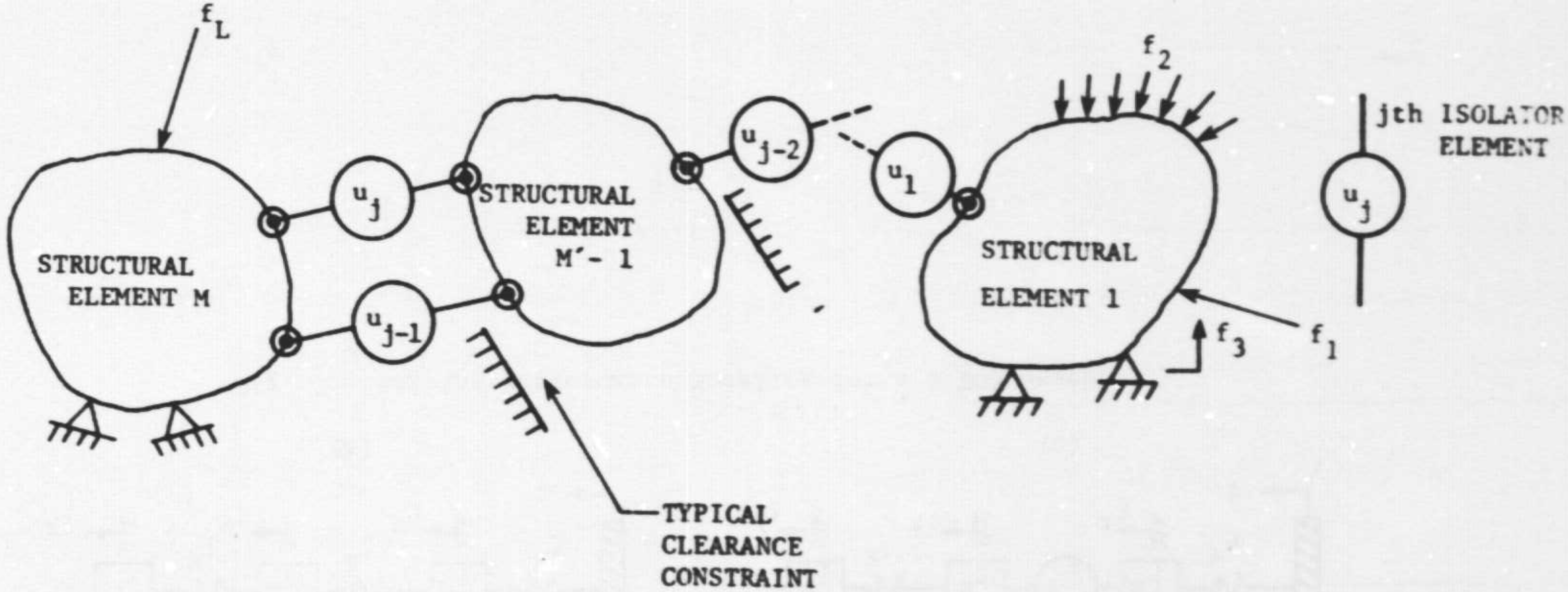

Fig. 1 - Structural Configuration for Determining Limiting Performance Characteristics

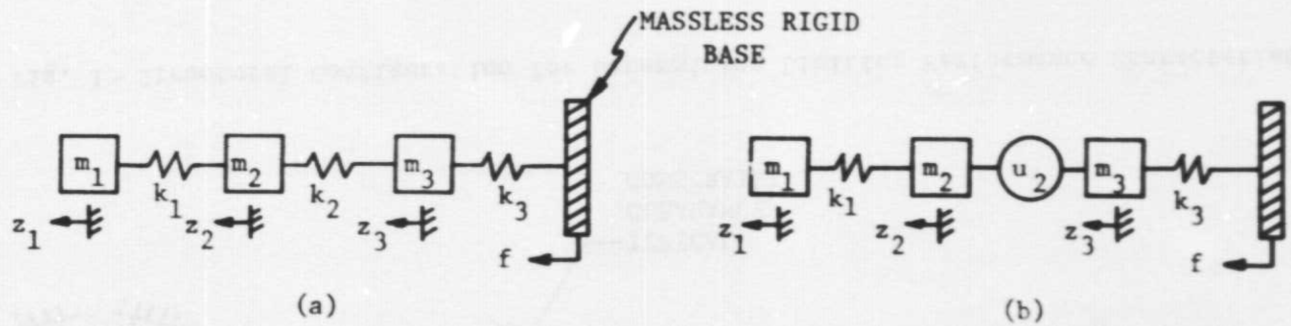

Fig. 2 - Limiting Performance Modeling for a 3 DOF System

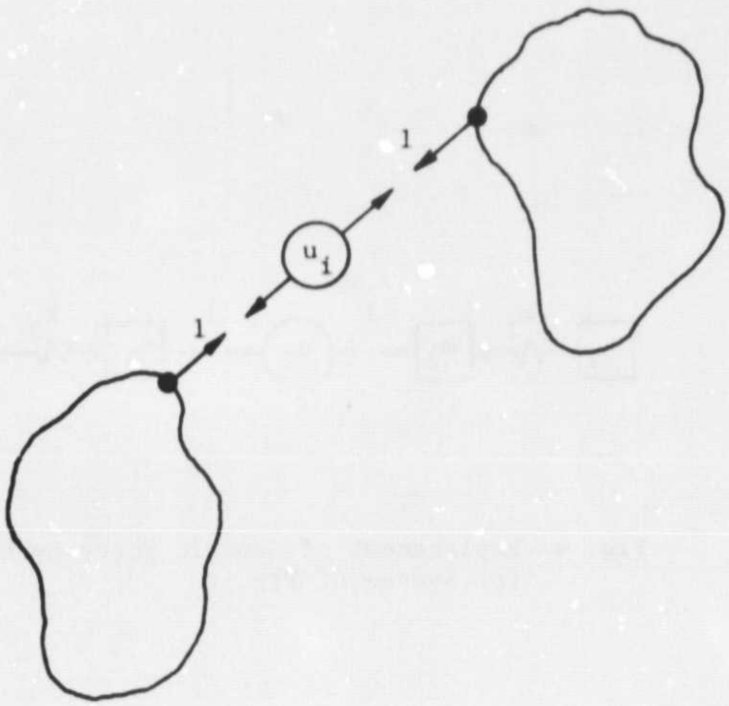

Fig. 3 - Replacement of Generic Force by Unit Loads

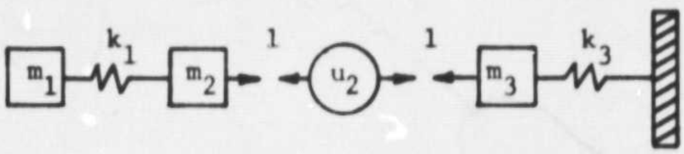

Fig. 4 - Replacement of Generic Force by Unit Loads for System of Fig. 2

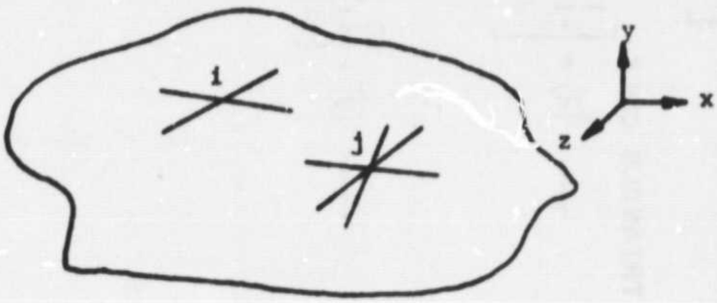

Fig. 5 - Configuration for Forming Constraints and Performance Index

```
MASS MATRIX(DIAGONAL ELEMENTS ONLY)
  .500          1.500          2.500

ASSEMBLY STIFFNESS MATRIX(UPPER TRIANGLE ONLY)
  10.000    -10.000
  10.000      0.000
  30.000      0.000

LOAD VECTOR
LOAD CASE 1 :
  0.000   -1.000    1.000
```

$$\longrightarrow [M] = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 2.5 \end{bmatrix}$$

$$\longrightarrow [R] = \begin{bmatrix} 10 & -10 & 0 \\ -10 & 10 & 0 \\ 0 & 0 & 30 \end{bmatrix}$$

$$\longrightarrow [V] = \begin{Bmatrix} 0 \\ +1 \\ -1 \end{Bmatrix}$$

Fig. 6 - SAP IV Generated Matrices for System of Fig. 2

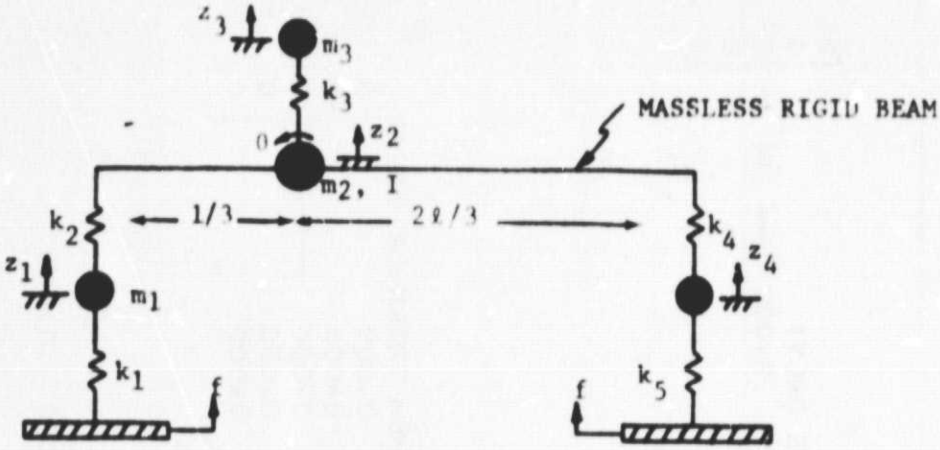

Fig. 7 Automobile Model

```
MASS MATRIX(DIAGONAL ELEMENTS ONLY)
 2.000    8.000    .500    1.000    2.000
```

$$\longrightarrow [M] = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 & 0 \\ 0 & 0 & .5 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

```
ASSEMBLY STIFFNESS MATRIX(UPPER TRIANGLE ONLY)
    3.000     -2.000      8.000      0.000
    8.000      8.000     -4.000     -2.000
  160.000      0.000    -16.000      0.000
    4.000      0.000      0.000      0.000
    3.000      0.000      0.000      0.000
```

$$\longrightarrow [K] = \begin{bmatrix} 3 & -2 & 8 & 0 & 0 \\ -2 & 8 & 8 & -4 & -2 \\ 8 & 8 & 160 & 0 & -16 \\ 0 & -4 & 0 & 4 & 0 \\ 0 & -2 & -16 & 0 & 3 \end{bmatrix}$$

```
LOAD VECTOR
LOAD CASE 1:
  0.000    0.000    0.000    0.000    0.000
```

Fig. 8 - Matrices for Model of Fig. 7

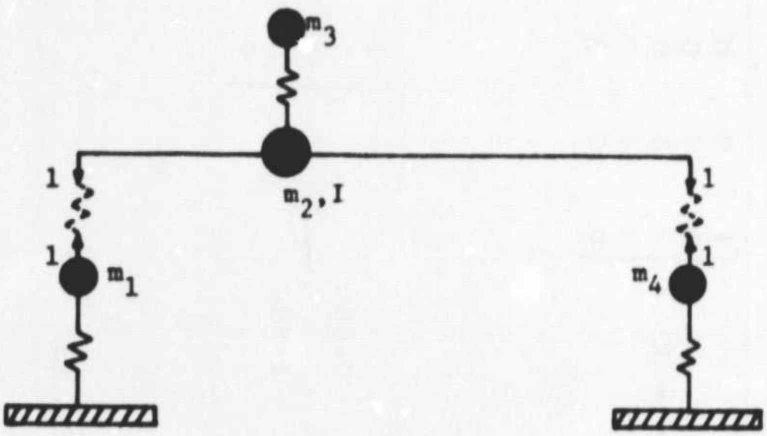

Fig. 9 Unit Forces for Model of Fig. 7

```
MASS MATRIX(DIAGONAL ELEMENTS ONLY)
 0.000    0.000    0.000    0.000

ASSEMBLY STIFFNESS MATRIX(UPPER TRIANGLE ONLY)

  1.000     0.000     0.000     0.000
  4.000     0.000    -4.000     0.000
   .000     0.000     0.000     0.000
  4.000     0.000     0.000     0.000
  1.000     0.000     0.000     0.000
```

$$\longrightarrow [\bar{K}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & -4 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -4 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

```
LOAD VECTOR
LOAD CASE 1:

   1.000    -1.000     4.000     0.000     0.000
LOAD CASE 2:
   0.000    -1.000    -8.000     0.000     1.000
```

$$\longrightarrow [V] = \begin{bmatrix} -1 & 0 \\ 1 & 1 \\ -4 & 8 \\ 0 & 0 \\ 0 & -1 \end{bmatrix}$$

Fig. 10 - Matrices for Eq. (3)